MR T NELSON
BOX 1546
POUGHKEEPSIE NY 12603

FOR:

more efficient, effective programming in 360/COBOL

new idea igniters (such as standards)

quick, authoritative answers to your technical questions

keeping abreast and implementing this major language the easy way

SEE:

SYSTEM/360 COBOL

by S.M. Bernard,
Systems Engineering Manager, IBM

BONUS FEATURES FOR YOU

. a unique "stand-alone" 37-page COBOL STANDARDS MANUAL section

. how to plan and structure COBOL programs

. step-by-step instructions for debugging COBOL programs

. considerations for Disk or Tape Operating System (including control cards and label formats)

. differences between DOS and Operating System/360

. programming for overlays (including an assembly language subprogram)

. coding subprograms (with examples)

VERSATILE APPLICATION

This new self-teaching manual gives you easy-to-apply hints, guidelines, groundrules, and programming conventions. It is oriented to Disk Operating (DOS) and Tape Operating (TOS) COBOL. And, most of the material is equally applicable to full Operating System (O/S).

SIMPLE TO USE, EASY FUTURE REFERENCE

The practical approach and step-by-step sequence makes it an ideal self-study guide for beginning programmers. You only need a basic knowledge of programming and some familiarity with high level languages. Whenever possible, the manual is written in outline form to allow for easy reference and use by more experienced programmers. Additional materials vital for effective 360/COBOL programming are included in a series of appendices.

QUICK ANSWER FEEDBACK

Arrangement of material is in a logical step-by-step sequence. For example --

Section I answers a few basic questions -- Why COBOL? -- How do you plan for programming? -- What is the structure of a COBOL program? -- What are the elements of a COBOL program?

Section II describes program identification, file identification, and file organization techniques as they relate to COBOL.

Section III is devoted to file, record and data descriptions.

Section IV concentrates on the statements used to code the body of a COBOL program.

Section V is a series of independent topics describing interfacing with DOS, COBOL Words and statement charts, debugging programs, Operating Systems considerations, and more.

Section VI is the "stand-alone" standards manual.

REMAINS USEFUL FOR YEARS: You'll use many of the practical examples of how to code in COBOL and refer to the standards manual daily as an idea sparker to make your work easier and more effective.

Furthermore, you get authoritative experience. The author, S.M. Bernard, has installed eighteen medium and large scale System 360's (both O/S and DOS COBOL as well as other 360 languages). As a Systems Engineering Manager for IBM, his job is to know how to meet the needs of System 360 users. In addition, Mr. Bernard is in frequent demand as a guest lecturer at many professional societies.

YOU ARE INVITED TO READ THIS BOOK, WITHOUT COST OR OBLIGATION

Send today for your FREE 15 day examination copy of SYSTEM/360 COBOL. When you see the detailed coverage in this manual, you'll be glad you ordered your copy! The examples alone will increase your programming skills as you never thought possible. If you're not completely satisfied, return the book and you won't owe a cent, not even for the expert know-how you'll surely get during the 15 day trial reading period. What have you go to lose?

Very truly yours,

Steven T. Landis

Steven T. Landis
for Prentice-Hall

TABLE OF CONTENTS

(TABLE OF CONTENTS cont'd. on back page 4)

Specimen Pages from the 37-page COBOL STANDARDS MANUAL section

(All greatly reduced -

actual page size is 8 1/2 x 11")

BLOCK DIAGRAMMING STANDARDS

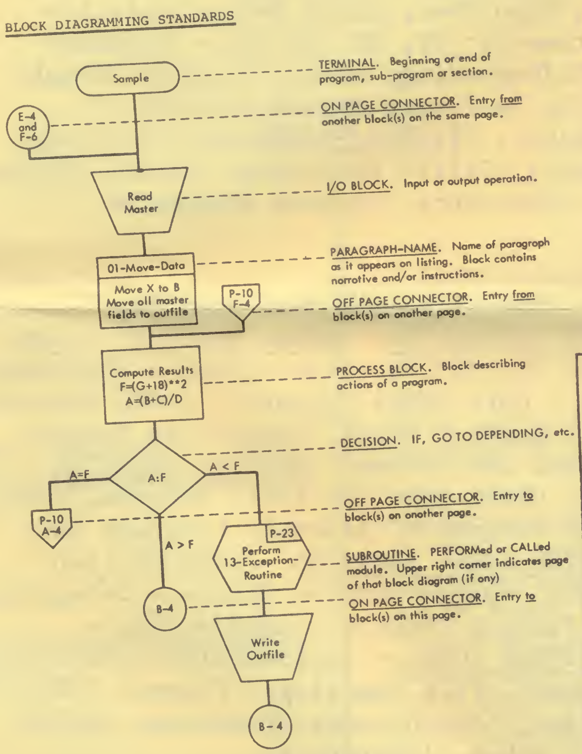

Sample
TERMINAL. Beginning or end of program, sub-program or section.
E-4 and F-6
ON PAGE CONNECTOR. Entry from another block(s) on the same page.
Read Master
I/O BLOCK. Input or output operation.
01-Move-Data
Move X to B
Move all master fields to outfile
PARAGRAPH-NAME. Name of paragraph as it appears on listing. Block contains narrative and/or instructions.
P-10 F-4
OFF PAGE CONNECTOR. Entry from block(s) on another page.
Compute Results
F=(G+18)**2
A=(B+C)/D
PROCESS BLOCK. Block describing actions of a program.
DECISION. IF, GO TO DEPENDING, etc.
A=F
A:F
A < F
P-10 A-4
OFF PAGE CONNECTOR. Entry to block(s) on another page.
A > F
P-23
Perform 13-Exception-Routine
SUBROUTINE. PERFORMed or CALLed module. Upper right corner indicates page of that block diagram (if any)
B-4
ON PAGE CONNECTOR. Entry to block(s) on this page.
Write Outfile
B - 4

Figure 60. Block Diagramming Standards

286

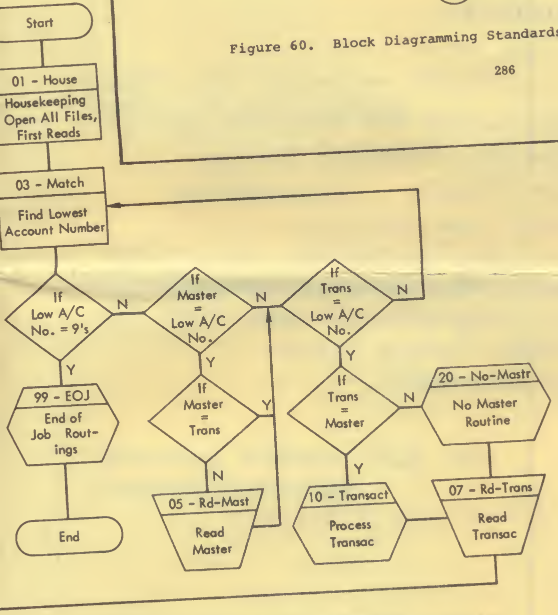
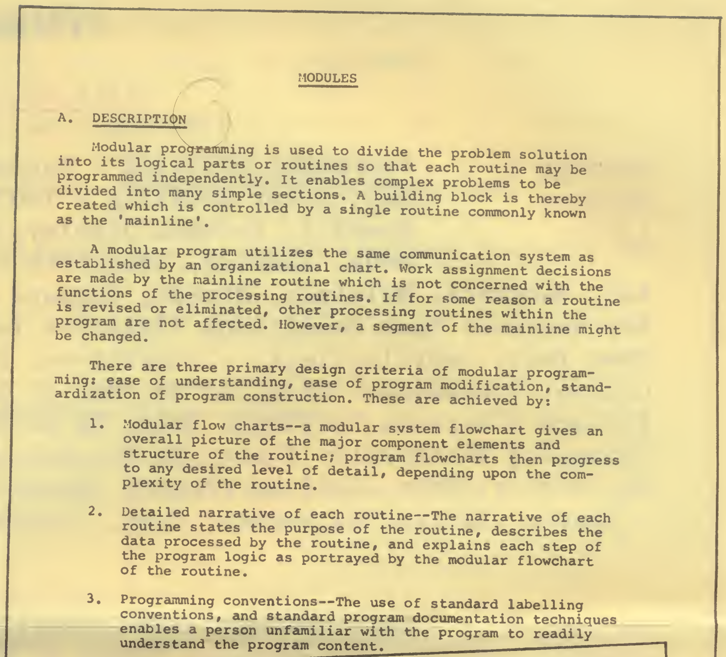


Figure 63. Main Line Block Diagram - Balance Line Approach

295



MODULES

A. DESCRIPTION

Modular programming is used to divide the problem solution into its logical parts or routines so that each routine may be programmed independently. It enables complex problems to be divided into many simple sections. A building block is thereby created which is controlled by a single routine commonly known as the 'mainline'.

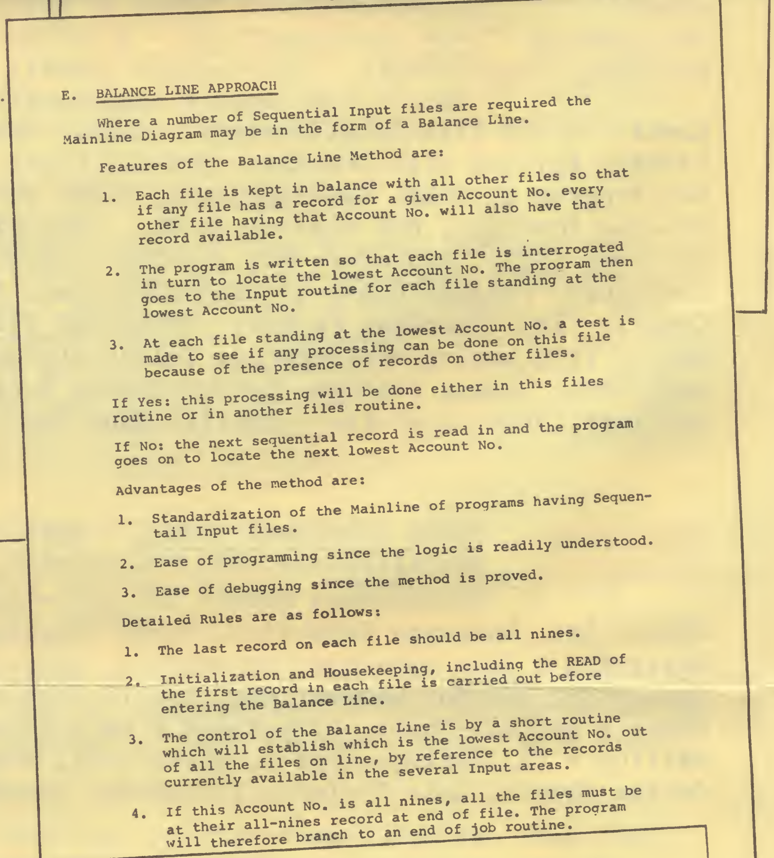
A modular program utilizes the same communication system as established by an organizational chart. Work assignment decisions are made by the mainline routine which is not concerned with the functions of the processing routines. If for some reason a routine is revised or eliminated, other processing routines within the program are not affected. However, a segment of the mainline might be changed.

There are three primary design criteria of modular programming: ease of understanding, ease of program modification, standardization of program construction. These are achieved by:

1. Modular flow charts--a modular system flowchart gives an overall picture of the major component elements and structure of the routine; program flowcharts then progress to any desired level of detail, depending upon the complexity of the routine.

2. Detailed narrative of each routine--The narrative of each routine states the purpose of the routine, describes the data processed by the routine, and explains each step of the program logic as portrayed by the modular flowchart of the routine.

3. Programming conventions--The use of standard labelling conventions, and standard program documentation techniques enables a person unfamiliar with the program to readily understand the program content.



E. BALANCE LINE APPROACH

Where a number of Sequential Input files are required the Mainline Diagram may be in the form of a Balance Line.

Features of the Balance Line Method are:

1. Each file is kept in balance with all other files so that if any file has a record for a given Account No. every other file having that Account No. will also have that record available.

2. The program is written so that each file is interrogated in turn to locate the lowest Account No. The program then goes to the Input routine for each file standing at the lowest Account No.

3. At each file standing at the lowest Account No. a test is made to see if any processing can be done on this file because of the presence of records on other files.

If Yes: this processing will be done either in this files routine or in another files routine.

If No: the next sequential record is read in and the program goes on to locate the next lowest Account No.

Advantages of the method are:

1. Standardization of the Mainline of programs having Sequentail Input files.

2. Ease of programming since the logic is readily understood.

3. Ease of debugging since the method is proved.

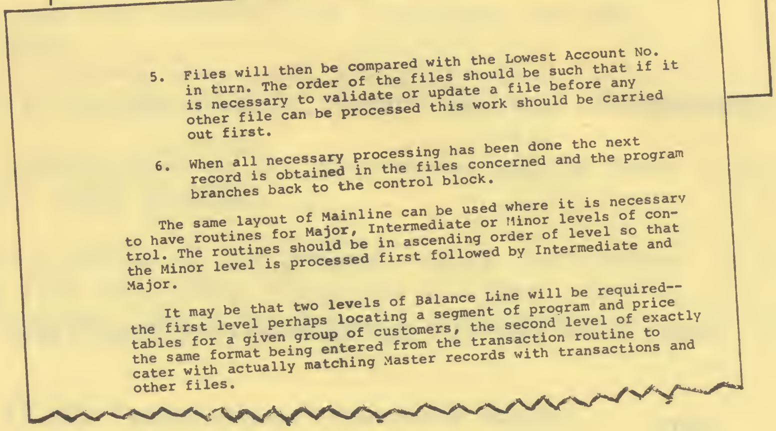
Detailed Rules are as follows:

1. The last record on each file should be all nines.

2. Initialization and Housekeeping, including the READ of the first record in each file is carried out before entering the Balance Line.

3. The control of the Balance Line is by a short routine which will establish which is the lowest Account No. out of all the files on line, by reference to the records currently available in the several Input areas.

4. If this Account No. is all nines, all the files must be at their all-nines record at end of file. The program will therefore branch to an end of job routine.



5. Files will then be compared with the Lowest Account No. in turn. The order of the files should be such that if it is necessary to validate or update a file before any other file can be processed this work should be carried out first.

6. When all necessary processing has been done the next record is obtained in the files concerned and the program branches back to the control block.

The same layout of Mainline can be used where it is necessary to have routines for Major, Intermediate or Minor levels of control. The routines should be in ascending order of level so that the Minor level is processed first followed by Intermediate and Major.

It may be that two levels of Balance Line will be required--the first level perhaps locating a segment of program and price tables for a given group of customers, the second level of exactly the same format being entered from the transaction routine to cater with actually matching Master records with transactions and other files.

SYSTEM/360 COBOL

TABLE OF CONTENTS CONTINUED

SECTION IV - PROGRAM LOGIC

PROCEDURE DIVISION - Introduction - Compiler-Directing Declaratives: Use, Error Handling - I-O Statements: Open, Close, Read, Write, Rewrite, Accept, Display, Exhibit - Data Manipulation Statements: Move, Examine, Transform - Arithmetic Statements: General Format, Add, Subtract, Multiply, Divide, Compute - Arithmetic Expressions - Conditional Expressions: Test Conditions (Relation Test, Sign Test, Class Test, Condition-Name Test, Overflow Test) - Conditional Statements: IF, ON (Count) - Sequence Control Statements: Stop, Go To, Use of Go To/Depending, Alter, Perform (Simple Perform & Perform Varying), Exit, Go To Within Performs, Nested Perform - Debugging Statements: Trace, On (Count), Exhibit, Display - Compiler-Directing Statements: Note, Calling Program (Enter Call), Called Subprogram (Entry, Return) - Source Library Statements: Copy, Include - Procedure Division Standards.

SECTION V - OTHER PROGRAMMING CONSIDERATIONS

DATA HANDLING CONSIDERATIONS - Miscellaneous Data Handling Consideration. TABLE OF COBOL STATEMENTS. COBOL RESERVED WORDS. DOS CONSIDERATIONS - Structure of DOS Libraries - Source Statement Library - Relocatable Library - Core Image Library - COBOL Control Card - Job Control Statements: Control Cards, Simplified Control Cards - S/360 File Label Formats - Sample COBOL Job Stream. OVERLAY HANDLING - Linkage Editing with Overlay - Overlay Logic: COBOL Main (or Root) Program, COBOL Sub-Program B, COBOL Sub-Program C, COBOL Sub-Program D, Assembly Language Sub-Program (Ovrlay), DOS Control Cards. DEBUGGING COBOL PROGRAMS - Introduction: Data Map (SYM), Procedure May (LISTX) - Typical Source Program Errors - Debugging Procedure: Program Check at Execution Time, Program Check at Compile Time - Compile-Time Debugging Packet. OPERATING SYSTEMS COBOL - Introduction - Environment - Programming Differences: Select Statement, File Handling - F-Level COBOL: Sort Verb, Report Feature, Corresponding Option, Asynchronous Processing (Multi-Tasking), Other Features - Overlay Handling - Other Differences.

SECTION VI - STANDARDS

COBOL STANDARDS MANUAL - Introduction - Preparation for Programming: Specification, Run Description, File Description, Standard Naming Techniques, Record Layouts - Application Flow Charts - Programming: Check List, Language Selection, Block Diagramming - Modules: Description, Designing a Run, Mainline Conventions, Components, Balance Line Approach, Processing Routine Conventions, Module Size - Labels & Names: Modules, Paragraph-Names & Section-Names, File-Names & Record-Names - General Programming Rules: Editing & Validating, Aids to Operations, Programming - COBOL Coding Rules: Coding Sheets, Data Division Standards, Procedure Division Standards.

Published July 1968 312 Pages 65 Charts, Instructions, Examples & Diagrams
Large 8 1/2 x 11" format, cloth-bound

Order your FREE 15 day examination copy from:

PRENTICE-HALL, INC.
Englewood Cliffs, N. J. 07632

DCOZ

Postage
Will Be Paid
by
Addressee

No
Postage Stamp
Necessary
If Mailed in the
United States

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 365, ENGLEWOOD CLIFFS, N. J.

PRENTICE-HALL, INC.

ENGLEWOOD CLIFFS, NEW JERSEY

07632

For More Efficient, Effective Programming in 360/COBOL;
Order Your Copy Of:

SYSTEM/360 COBOL

by S. M. Bernard

Yes, I would like a FREE 15-day examination copy of this handy self-teaching manual. When I see the easy-to-apply hints, guidelines, groundrules, and programming conventions it contains, I'll send you my check for $9.95 plus postage and handling. Otherwise, I'll return the book within the 15-day period and not owe a cent.

Name____

Address____

City______________________State______________________Zip__________

SAVE! If payment accompanies order, publisher pays postage and handling. Same return privilege, refund guaranteed.